Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-92-06

Main Topics of DAI: A Review

Achim Schupeta

March 1992

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

Postfach 20 80
D-6750 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum
für
Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988 by the shareholder companies Daimler-Benz, IBM, Insiders, Fraunhofer Gesellschaft, GMD, Krupp-Atlas, Digital-Kienzle, Philips, Sema Group Systems, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ❏ Intelligent Engineering Systems
- ❏ Intelligent User Interfaces
- ❏ Intelligent Communication Networks
- ❏ Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

Main Topics of DAI: A Review

Achim Schupeta

DFKI-RR-92-06

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-9104).

MAIN TOPICS OF DAI: A REVIEW

Achim SCHUPETA

DFKI (German Research Center for Artificial Intelligence)
Research group AKA-MOD
Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11
GERMANY

email: schupeta@dfki.uni-sb.de

A new branch of artificial intelligence, distributed AI, has developed in the last years. Topic is the cooperation of AI-systems which are distributed among different autonomous agents. The thereby occuring problems extend the traditional AI spectrum r DAI-relevant topics: Knowledge representation, task-decomposition and -allocation, interaction and communication, cooperation, coordination and coherence, organizational models, agent's modelling of other agents and conflict resolution strategies (e.g. negotiation). First we try to describe the role of DAI within AI. Then every subsection will take up one special aspect, illuminate the occurring problems and give links to solutions proposed in literature. Interlaced into this structure are sketchy descriptions of a few very prominent and influential DAI systems. In particular we present the Contract Net Protocol, the Distributed Vehicle Monitoring Testbed, the Air Traffic Control problem and the Blackboard Architecture.

Contents

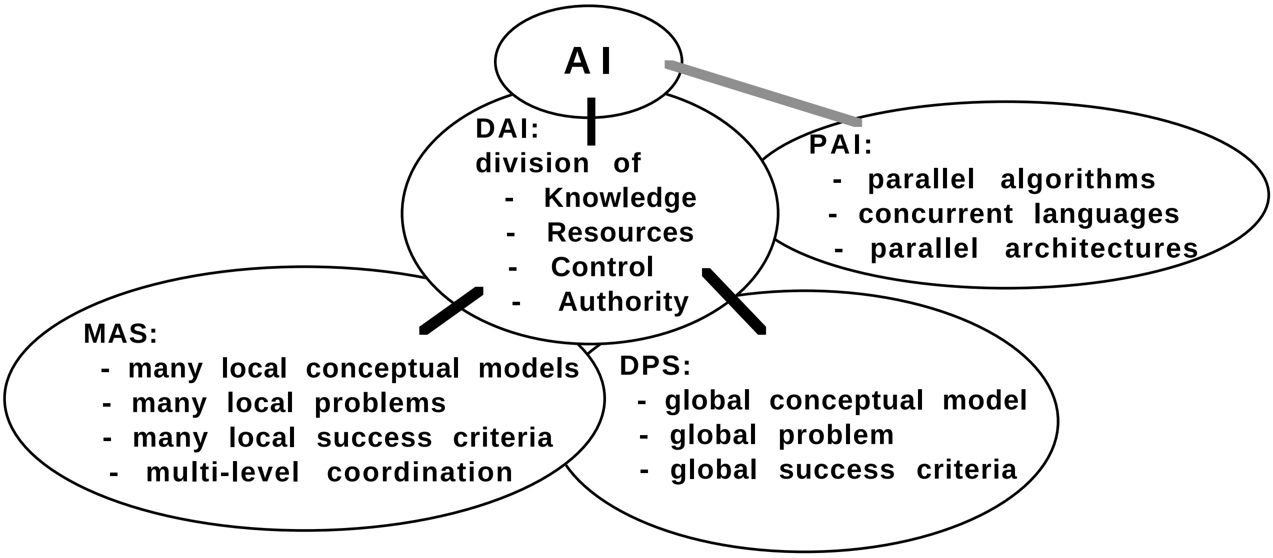
1. Multi-Agent Systems and Distributed Problem Solving

In the last decade a new branch of AI has developed out of an increasing interest in concurrency and distribution. It was named **Distributed Artificial Intelligence** (DAI) and it is concerned with problem solving by a society of agents. Each agent is autonomous, which means it has its own behaviour, planning strategies, goals, skills and so on. Though the idea of autonomous cooperating rational agents in a multi-agent society is relatively new and far away from coming up with well understood results, (there are more open questions than appropriate answers; see (Coe88)) there is an increasing amount of literature (Huh87, BG88, DM90, DM91) and the area is still growing rapidly. The availability of multicomputers, of networks with many connected processors and new developed concurrent languages in alliance with an increasing interest in solving large real world problems, that involve multiple knowledge domains, locally limited resources and bounded rationality in an asynchronous continuously changing setting form the major motivations for the DAI research direction. Arm-length relationships and perceptual inconsistency between local autonomous but interdependent nodes are also characteristics for typical DAI problems. Take for example a worldwide banking system: It is in principle not possible to have a complete global view of it and it is running day and night in a dynamically changing environment. The goal of DAI research is to build an accurate conceptual model of natural and social systems with the above mentioned typical problem characteristics. Furthermore, other disciplines related to AI become more relevant e.g. mathematical game theory, organizational theory in economics, sociology, linguistics, psychology and cognitive science to mention some of them. Assuming that <u>intelligence is intrinsically no stand-alone concept</u>, but can only be developed within a community, DAI research is interested in the cooperative, organized interaction of several agents and has to take into consideration that the reductionistic viewpoint that agents form a society has to be extended by the reverse aspect that a society also forms the agents. To characterize DAI a little bit more precise, we will try to sketch their position within AI and introduce some usually used terminology.

The following ideas reflect the tenor of (BG88):

Distributed artificial intelligence (DAI) is the subfield of AI, that is concerned with *concurrency* in AI at many levels. DAI branches into the area of ***Distributed Problem Solving*** (DPS), which considers how the work of solving a particular problem can be divided among a number of cooperating and knowledge-sharing modules or nodes, on the one hand and into ***Multi-Agent Systems*** (MAS) on the other. In MAS the coordination of intelligent behaviour of a collection of autonomous intelligent agents is the main concern. Coordination of knowledge, plans, different skills and actions itself is a process, that the agents have actively to reason about.

While both of these areas have to deal with the division of knowledge, resources, control and authority, in DPS we usually have a global conceptual model, global problems and a global success criterion, where in MAS we have many local conceptual models, problems and success criteria. A third area - Parallel Artificial Intelligence (PAI) - stresses more on performance problems than on conceptual advances and is involved in the development of parallel computer architectures, parallel languages and algorithms. But a sharp distinction between these areas could not be drawn as there is still no clear commonly accepted definition of the term “autonomous cooperating agent”. The following figure sketches this structure:



Sometimes the term “Cooperative Distributed Problem Solving” (CDPS) is used for DPS. Edmund H. Durfee pointed out the following aspect of distinction:

"In CDPS (and DPS), the agents are faced with problems to solve (goals to achieve) that are beyond their individual capabilities, and so must work together to satisfy their goals. In MA systems, the agents are not assumed to be facing problems beyond their individual capabilities, but because they share an environment, they still must deal with issues of coordinating to resolve conflicts between actions or to take advantage of the actions of others in achieving their goals. In my opinion, the essential characteristic dividing the areas is not whether the agents have common or distinct goals, but whether or not the goals in whatever form require collective activity."

Micheal Huhns sees an explicit representation of the goals as a necessary condition for an agent:

"I distinguish a DPS system from a Multiagent System as follows: a DPS system can consist of multiple expert systems or multiple knowledge sources (KSs), as in a blackboard-based system. I think of agents as having an additional characteristic--an explicit representation of their goals. The conventional description of an agent requires the abilities to perceive, reason, and act. Adding the requirement that goals be represented enables these goals to then become a subject for their reasoning. To me, the lack of this in most expert systems is what precludes those systems from being thought of as agents. For example, a thermostat (which is like an expert system with one rule, and which perceives its environment, reasons about it, and acts) just doesn't seem to be an intelligent agent, because it can't reason about what it is doing--maintaining some temperature."

Another possible dimension of DAI is the *adaptivity* of a system (BG88, chapter 1), which means to ask how flexible does the system react to changes in the problem or in its environment:

> PAI usually is only adaptive to *temporal* uncertainty, while DPS can handle uncertainty in the *problem solving knowledge*. MAS should be able to handle changing *problem solving roles* and even actively construct their coordination framework.

(BG88) further pointed out, that DAI is engaged in *large-grained coordinated intelligent systems* in contrast to the connectionistic approach recently mainstreamed by (McCR87), where the explanation of higher-level mental functions on the basis of highly parallel collections of very simple computing elements is the main concern.

An exact definition of “autonomous cooperating agent” cannot be given, because the terminology in this area is not as clear as it should be. Depending on the complexity of the system and the author's taste the terms *node* or *module* are sometimes used as synonyms for agent. (BG88) spoke of an agent as a computational process with a single locus of control and/or “intention”, but they conceded, that this could not be treated as a definition. To give an introduction I will sketch the main problems and topics of DAI research interleaved by links to the solutions proposed in the associated literature. Due to the richness of different aspects and problems most of the existing DAI systems focus on only a few of these problems. But they all show, that the distribution of knowledge and behaviour as a new paradigm brings up a different promising view of problem solving and justifies DAI as a research area worth to be developed further with more research power.

In the following sections we will glance over current DAI research along the topics of knowledge representation, task-decomposition and -allocation, interaction and communication, cooperation, coordination and coherence, organizations, modelling other agents and the problems of conflict resolution and negotiation. A few prominent DAI research domains will be introduced in interspersed subsections. The intention is to introduce the terminology and major problems of DAI, solutions are rare anyway in this young research direction, and to refer to some interesting and fundamental research papers.

2. Knowledge Representation (KR)

As in classical AI systems the problem of an appropriate representation of an agents knowledge occurs also in DAI. Knowledge comprises **descriptive knowledge** of an application-domain as well as **procedural knowledge** in form of algorithms, rules and operators. Theoretical foundation for almost every KR-formalism is the *predicate calculus*. The first step in modelling a domain is to identify and name the relevant objects, their properties and relations to other objects. Which operations can be performed on these objects and how do they change the relations between them? Conventional *planning systems* model their world as a data-base of logical formulas, that describe the world state and represent operators as sets of preconditions and the associated world changes as add- and delete-lists. STRIPS (FN71) can be viewed as a kind of “great mother of all planners” using this approach. An overview of the development of planners is given in (Cha87).

Many aspects of real world problems cannot or not efficiently be modelled with classical logic. E.g. an axiomatization of natural numbers cannot be done in first order logic. Statements about sets and relations of sets, about temporal relations, statements, that introduce some uncertainty or probability or statements about wishes, beliefs, intentions and so on, can be expressed shortly and elegantly in natural language but only ambiguously and awkwardly if at all in the predicate logic formalism. Several approaches have been undertaken to tune up the classical logic to overcome such difficulties. Modal operators were used to model knowledge and belief or temporal aspects of a domain. Mechanisms to model uncertainty often use an additional parameter associated with each formula to represent some degree of certainty or vagueness. The propagation of these values is problematic when knowledge is combined to infer new knowledge. Semantic nets, frames and constraint nets are special forms of representation that try to take advantages out of a structure that they impose on the knowledge. These approaches led to the development of KR-languages like KL-ONE, where knowledge is organized in hierarchies or nets with various possibilities to associate different nodes, so that the contextual embedding of knowledge into other knowledge can be supported better. Confer (Ric89) for a detailed and comprehensive discussion of these problems and (BS85) for an overview of KL-ONE.

All these methods have in common, that they use an **explicit** or symbolic **representation** of knowledge: They need a formal language often equipped with some structure creating constructions and some rules to make inferences. (GN87) gives a good presentation of this logic-based view. Although this direction of AI research had an enormous impact on computer science, (Prolog and declarative programming, expert-systems, object-oriented programming ...) it is argued by some researchers, that an explicit KR is not an adequate model for human intelligence and will never be capable to catch the notion of common sense reasoning.

In DAI we can distinguish two different lines of research:

-> The ***logic- and knowledge-based approach*** relies on an explicit symbolic representation, where agents reason via *logical deduction*.

-> The ***behaviour-based approach*** emphasizes the necessity of a *holistic point of view*.

The logic-based approach stays in the tradition of well known logic-based AI. The idea is straightforward: Every agent is supplied with a knowledge base and an interpreter to apply some rules and thus is a little AI system in the classical sense. Additionally each agent needs some further concepts, for example it must have *communication facilities*, methods to install *cooperation* with other agents and the faculty to perform some *meta-reasoning*, e.g. maintaining *introspective knowledge* and models of other agents or resoning about *beliefs* and *intentions* (cf. (CL90) for a logic of intentions.). So this research line is engaged in the development of logics for an appropriate representations for these concepts. This means, that the representation must be compatible with the chosen basic representation and must catch the semantics of the modelled concept. A few more words about that will be said in the section about agent modelling.

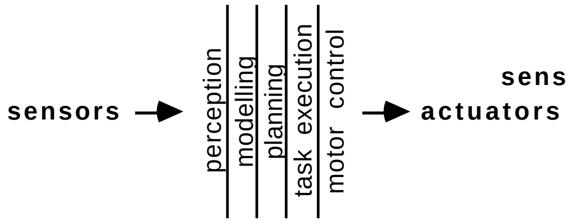
The behaviour-based approach grounds on the supposition that an explicit logic-oriented representation ends up in an impasse when one tries to model common sense knowledge with it. Common sense behaviour of humans or animals is not driven by logical deductions or inferences and thus a modelling within that paradigm is not adequate. The behaviour-based approach thus demands for a non modular *integrated agent architecture*, e.g. the kind like it was proposed by Brooks in (Bro86). His **subsumption architecture** fits well to the requirements of the behavioral approach especially to respond very fast to unpredictable events and to classify the required behaviour along a dimension of more important - less important:

Subsumption Architecture

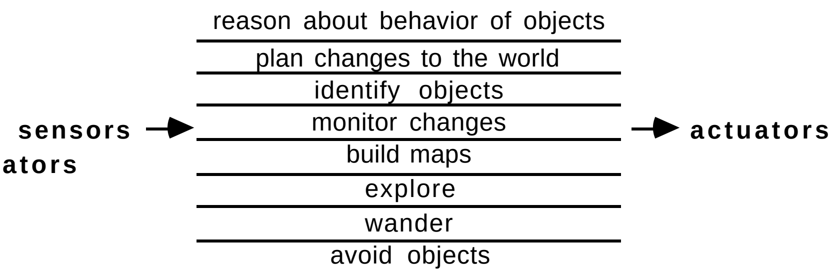
The goal of Brooks and his research group is to build cheap and simple real-world (not only simulated!) autonomous intelligent mobile robots, which show a kind of insect-like behaviour. Previous approaches to that problem had used a horizontal decomposition into vertical slices. That means: The robot's control system has the problem to connect the input (sensing) with the output (acting). The usual approach was to decompose that problem into subproblems like "perception", "mapping sensor data into a world representation", "planning", "task execution", "motor control" ... or a more finer decomposition. The problem with that kind of decomposition is, that an instance of each piece must be built in order to run the robot at all. Brooks' idea was to decompose the problem vertically into different **task achieving behaviors**. He defined several *levels of competence* as for example:

0.) Avoid contact with object (whether they are moving or stationary).
1.) Wander aimlessly around without hitting things.
2.) Explore the world by seeing places in the distance that look reachable and heading for them.
3.) Build a map of the environment and plan routes from one place to another.
4.) Notice changes in the static environment.
5.) Reason about the world in terms of identifiable objects and perform tasks related to certain objects.
6.) Formulate and execute plans that involve changing the state of the world in some desirable way.
7.) Reason about the behaviour of objects in the world and modify plans accordingly.

Each level includes the previous level as a subset or to put it the other way, imposes additional constraints upon the behaviour. Each level connects specific inputs to some output. Lower layers can run totally independent of each higher level layer, which subsumes the lower level behaviour (hence the name subsumption architecture).



Traditional Decomposition



Decomposition along task-achieving behaviors

The development of a robot using this architecture is incremental: First a zero-level robot is built and debugged thoroughly. Then the level one control layer, which is able to examine level zero data and permitted to inject data into the internal interfaces of level zero is built. Each construction step ends with an autonomous independent robot, that shows an additional aspect of behaviour for each new level. The individual layers are implemented as simple networks of modules, which are *finite state machines*. Each module has a number of input lines with single element buffers and several output lines. Layers are constructed via wires that connect output lines of one module to input lines of another. Internally each module has several states and has access to a variable capable to hold LISP data structures. The next higher layer may be connected to this network via *suppression* or *inhibition*. A suppression wire from the higher level ends at an input line of a lower level module and can give an input signal to that module while suppressing the usual input for a predefined time. An inhibition wire just ends at an output line of a lower level module and a signal on that wire inhibits any usual output on that line for a defined time. The modules of all layers run completely asynchroneous and there is no more sophisticated mechanism to prevent the messages from beeing lost, (handshaking or acknowledgement) which actually happens quite often. Furthermore, there is no global shared memory or central control module. Brooks succeeded in building a few robot-creatures according to these simple design rules, that run around in the laboratory and office areas in the MIT AI-Laboratory and operate with people walking by, people deliberately trying to confuse them , and people just standing by watching them. In (Bro86) is a detailed description of the concepts of the subsumption architecture and in (Bro91) you can find a report of the ongoing research together with a very violent criticism of conventional (representation-oriented) AI research.

To end this short excursion about the subsumption architecture: Another aspect of the behaviour-based approach is the *embedding* of the agents into their environment. In the logical paradigm each agent *acts* as a consequence of a possibly large process of reasoning, which may take into consideration the agent's intentions, goals, its internal state, knowledge about the actual and historical world state and so on. All this knowledge
7 is usually represented explicit and must be maintained in the course of the actions. In the behavioral approach this is much simpler: An agent merely *reacts* in response to

environmental changes[1]. This is termed as **reactive behaviour** or a **situated action** (Suc87). The agent *copes* with the unpredictable events, that the environment constantly throws up. A more abstract and complex problem solving behaviour should emerge out of a series of abstractions of simple and fundamental reactive behaviour. There is not necessarily an explicit representation or internal world model within an agent. E.g. beliefs of an agent are represented implicit by its behaviour. For example an agent putting up its umbrella can be said to belief, that it is raining. There is no sentence in its database of the kind “belief(raining)”. Using ***the world as its own model*** could be taken as the motto of this implicit representation, and thus *perception* is much more important in the behavioral paradigm than it is in the logical paradigm. A good overview of the concepts, that build the basis of the behavioral approach is given in (CSW88). The following figure opposes the two approaches:

behavior-based

- **acting**
- **perception**
- **implicit representation**
- **situation-action-rules**
- **reactive behavior**
- **reactive acting without planning (situated actions)**
- **embedding into the environment**

logic-based

- **planning**
- **reasoning**
- **explicit representation**
- **formal logic calculus + knowledge base**
- **planned behavior**
- **acting after planning**
- **abstracted away from the environment**

But none of these concepts should be treated as a dogma and thus it seems to us, that the most fruitful research is that, where the best ideas of the two approaches are combined. We will give three examples of successful combinations in the rest of this section:

For example in (Mae89) the representation of *classical planning system operators* with preconditions, add- and delete-lists is combined with a dynamic and reactive mechanism to select the next operator for execution. This mechanism is based on the *spreading of activation* via links, that connect formulas from add-lists with same formulas of preconditions of other operators. Agents (= operators) with fulfilled preconditions and maximum activation will be executed. The actual state of the environment is the set of all true formulas and influences the behaviour as the source of spreading activation. Thus classical operator representation is combined with an implicit plan representation by a reactive action selection mechanism.

[1]Corresponding to the constructivistic theory of cognition, Hußman (Huß91) terms these changes as perturbations and treats them as triggers for the agents reactions.

The work of Hultman, Nyberg and Svenson (HNS89) combines an explicit KR within EETL, which stands for *Executable Explicit Temporal Logic*, with a *layered architecture* in a broad sense similar to Brooks subsumption architecture. The analyze layer receives goals or tasks, that a special decision procedure transforms into a set of well-formed EETL formulas, which represent a plan. The rule layer uses this data-base to create different types of rules, that are used to control actions, e.g. one type of rule determines the occurrence of a time point and another type modifies the set of actually executed actions. The process layer uses this rules to create several processes, that execute actions. This layer is subdivided into three sublayers and makes use of a state vector, that reflects all necessary information about the environment and the control system.

N. Seels describes in (See91) an approach to reconcile the logical approach to *model knowledge and beliefs as modal operators* with the behavioral approach of *reactive systems*. After analyzing previous logical approaches to the problem of logical omniscience, he introduces a relative simple reactive agent, that learns to respond correctly to incoming stimuli from the environment by maintaining possible histories as long as the agent is able to predict the future environment stimuli. It has now learned the environmental rules. A proper formalization of the possible histories over time leads to a replication of the Kripke semantics underlying a logic of knowledge and time. His conclusion is, that the agent satisfies the logic without explicitly using it.

Such approaches sound fruitful to reconcile the advocates of logic formalisms and those, who prefer the reactive behaviour viewpoint.

3. Task-Decomposition and Task-Allocation

One of the main problems of DAI systems could be formulated in form of the question:

➪ "Which agent does what task when?"

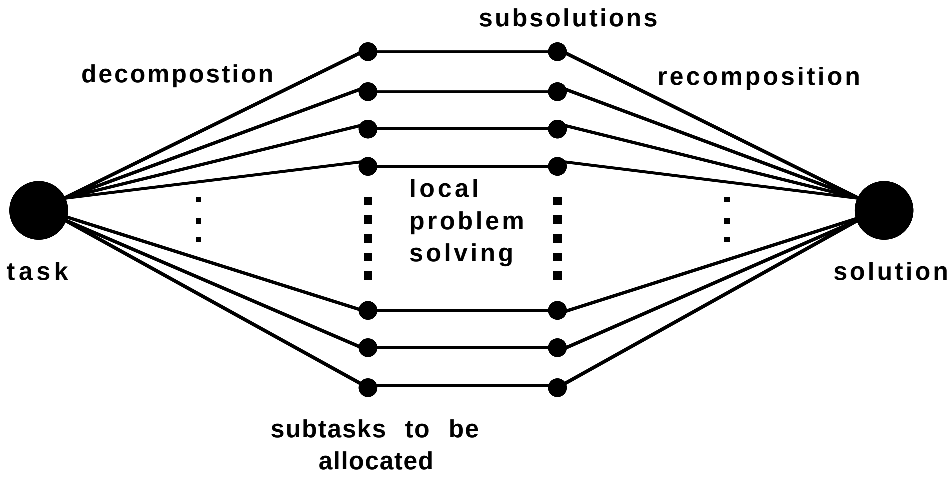
The basic model in a typical DAI system is to have a task, that is dividable along certain **decomposition criteria** into a set of subtasks. These subtasks are allocated according to some **allocation criteria** to different agents. Each agent solves its subtask and the results from the agents can be **recomposed** to a global solution. Several decomposition criteria have been suggested like e.g. *location* (spatial, temporal, logical), *abstraction* and "*interest area*" by Lesser and Erman (LE80). In general:

The task description should allow different alternative possibilities of task decomposition.

Lets consider a STRIPS-like representation for an example of this: A set of predicates defines the relations of the domain. Constant symbols represent the objects and each world state is represented by the set of all actually holding relations. Operators are described by preconditions, add- and delete-lists. One can think of the following possibilities for task decomposition (cf. Eis91):

- -decomposition by goal atom: One agent for each atom of the goal scene description.
- -decomposition by constant symbol: One agent for each constant symbol, i.e. for each object of the domain.
- -decomposition by predicate symbol: One agent for each predicate symbol, i.e. each agent is a specialist for installing a specific relation.
- -decomposition by operator symbol: One agent for each operator, i.e. each agent is a specialist for the execution of an operation.

In MAS with heterogeneous agents, the problem of task decomposition is enriched by the need to match resources and capabilities of different agents with appropriate tasks, so that the decomposition implies the allocation to a certain degree. The following figure depicts the general scheme of decomposition and recomposition:



Usually several constraints on *activities*, *available knowledge*, *time* or other *resources* restrict the space of possible decompositions. Little work in DAI has addressed an automated problem formulation and decomposition. So most decomposition processes are directly determined by the descriptions of available operators, and thus by the designers care. Not necessary to mention, that in general the representation of knowledge and the conceptual models of an agent have a very big influence on the decomposition process. Key issue for all decompositions is to find *dependencies* and logical *groupings* in the problem task and knowledge. Such groupings could for example be made on the basis of the *level of abstraction*, of *control-* and *data-dependencies*, the availability of *coordination* and the need for *interaction*. Malone (Mal87) suggests a **functional / product division** according to the organization model of companies in economy.

Another aspect of task decomposition is the tradeoff between *efficiency* and *reliability*. Decomposing a problem into redundant subtasks will make the results more reliable, will avoid uniqueness, and therefore will avoid possible bottlenecks, and is also a feasible solution to handle uncertainty. But possible disparities between the solutions of redundantly working agents could lead to more expense of communication and negotiation, when the solution is recomposed.

In most cases the possible decompositions are already built into the programmer-generated action description, which depends on the task representation and the task itself. The allocation of subtasks to agents often follows from the decomposition, especially in MAS with heterogeneous agents. A subtask is then allocated to the agent who provides the best fit to the task specification. Bottleneck avoidance, handling of uncertainty and reliability by redundant allocations, interaction dependencies or resource needs for special tasks may also drive *allocation decisions*.

Methods of task allocation often reproduce mechanisms used in human society. One prominent example is the Contract Net Protocol, which is introduced in the following section.

The Contract Net Protocol

The **Contract Net** system (Smi80, DS83) employs a *market-like allocation mechanism.* A task is *announced* by a manager to a set of agents, which then give their *bids* for this task, reflecting their interest that depends on their capabilities and actual loading. The manager selects the winning bid, *awards* a contract and monitors the contractor's performance by interim and final *reports* from the contractor. The manager has the possibility to terminate the contractor by a termination message at all hours. This could be useful when a task redundantly was given to several contractors.

Also a *recursive allocation* is allowed in the Contract Net system: If a task is too large to be performed by a single agent, the contractor splits it into several subtasks, announces them to find appropriate subcontractors, and takes the role of a manager itself. Thus the allocation work itself is decentralized and each node can play a dual role by being contractor in a contract and simultaneously manager in other contracts.

Focused addressing can be used, when the manager knows, that a subtask can only be performed by a few specialized agents. Then the manager does not broadcast the announcement, but selectively announces this task only to the specialists.

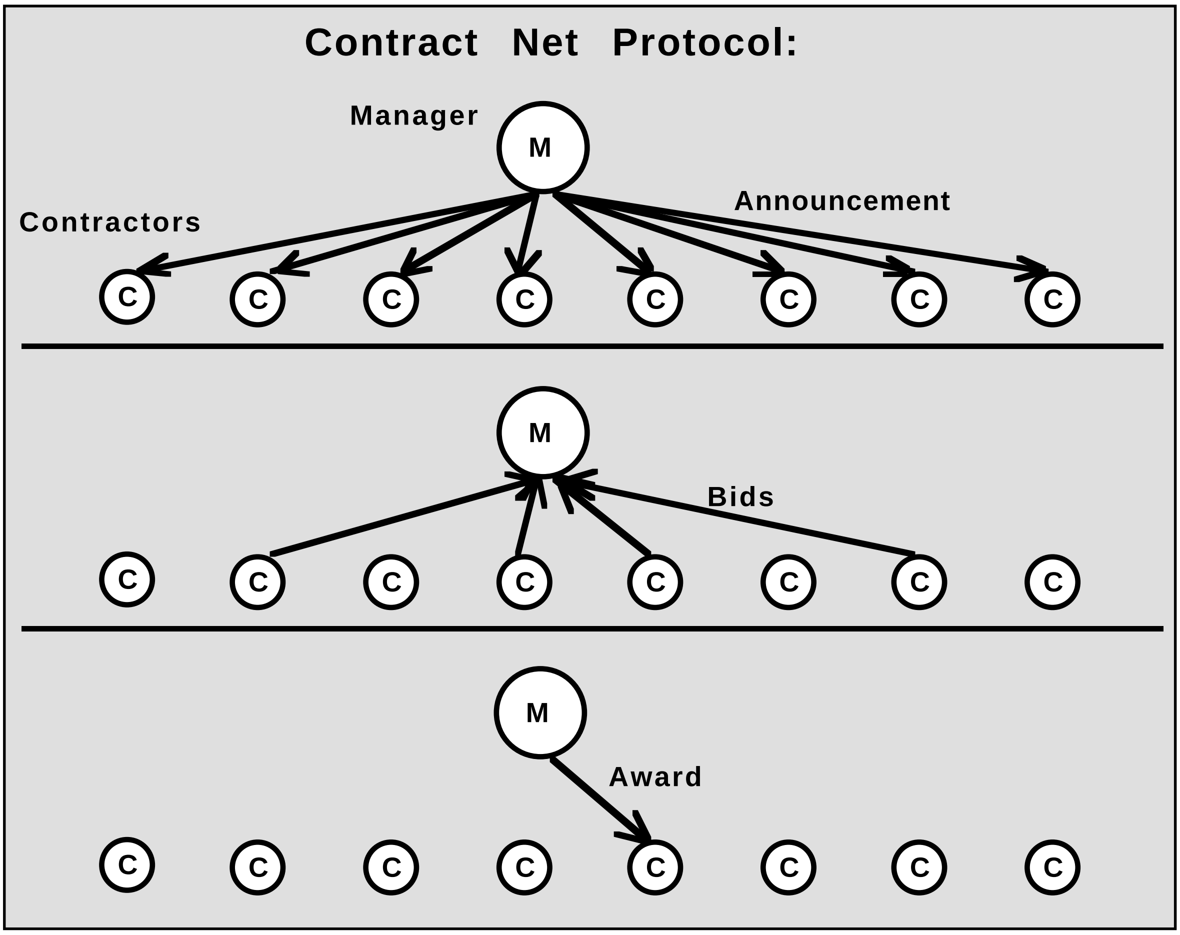
A high computational load in the net may lead to a *reversal of the normal negotiation protocol*: A *node-available message* allows an idle node to indicate, that it is searching for a task. This makes the protocol load-sensitive: While under normal conditions, when tasks are scarce, an auction is an important event, under heavy load conditions the availability of a node is an important event.

Note that in the usual way of processing, the contract is based on a *mutual selection*: The contractor can select from several announced tasks and the manager can select from several bids. Task announcements contain some *eligibility specification*, an abstract form of a *task description* and a *bid specification*, which specifies what information about the contractor must be given in a bid to enable the manager to compare the bids for its selection. A further slot of a task announcement specifies the *expiration time* of the task.

Every message contains the address, the sender, a type identifier and a contract-number in the header. The other information slots differ according to the message type. The information that fills these slots is expressed in a *common internode language*, which is built around a very simple attribute, object, value representation. There are a number of predefined (domain-independent) terms and also domain-specific term. It is assumed that all nodes understand this language. To summarize the main features of the CNET:

- We have a *two-way transfer of information* provided with the message types task announcement and bid.
- We have a *mutual selection* when a contract is assigned. Both partners have to make a choice.
- All selections are based on the nodes *local evaluation* of the gained information.

The following figure sketches the Contract Net's usual procedure:



The task allocation in a CNET-guided scenario is not necessarily optimal: First there is the *problem of timing*. Due to the asynchroneous processing of the nodes it may happen, that an idle node bids for a task, when a few seconds later it receives the task announcement of a much more lucrative task. Second there is the *problem of locatity*: The selection of a node is only based on its actual load, its capacity and the information gained by the task announcements or bids. It may happen that two managers compete for a specific idle node without knowing about this competition, and that at last one of them contracts with that node having an only slighly worse alternative in contracting with another node, while the other manager has to contract at last with a very much more worse other node. The globally better solution then would have been to couple the second manager with the idle node.

Another possible answer to the basic DAI question is to install predetermined or long term fixed *organizational roles* for the different agents. Each agent then has a restricted area of competence and will perform only tasks, that fit into its role. A voting mechanism could be employed to allocate tasks of special importance, e.g. in case of loss of a central planner or coordinator agent, a successor may be determined by voting. The term "***multi-agent planning***" refers to a scheme, where agents are treated as specialized resources. A single centralized planner or a few distributed planners generate plans for these agents.We will consider multi-agent planning and organizations in more detail in the sections 5. and 6.

4. Interaction and Communication

Interaction between the agents and between an agent and its environment is the essence of DAI, because the effects caused by interactions make a DAI system more than the sum of its components.

➪ <u>Interaction builds the foundation of communication, of cooperation and coordination, of the installation of an organization, and is the basis, that every representational model of other agents has to rely on.</u>

Ideally each interaction should cause a revision (or refinement) of the model of the other agent. Bond and Gasser (BG88) define interaction as follows:

> "We shall use ***interaction*** to mean some type of collective action in a MAS or DPS system , where one agent takes an action or makes a decision that has been influenced by the presence or the knowledge of another agent."

Presumably people from the behaviour-based approach to DAI would prefer to extend this definition to include the *interaction between agent and environment*, because they see the agent's actions as reactions to the environment's actions or let us better say the environmental changes. Each action can be viewed as an interaction between the agent and its environment. In the extreme case from the agent's viewpoint other agents are also a part of the environment. The complexity of an agent's behaviour is therefore also partly assigned to the environment. This aspect gets vivid by Simon's parable of the ant (Sim81):

> The ant making its way across an undulating and pebble-strewn beach attempts to pursue a straight-line course for her home, but has to cope with local gradients

and obstacles. The complexity of the ants behaviour is largely a reflection of the environments complexity.

An interesting question is:

Can interaction take place without communication?

The answer depends on the definition of communication. For example, the escaping behaviour of an antelope at the sight of a lion. The interaction is triggered by the simple act of perception of the antelope and motivated by implicit knowledge of the antelope about the lion's intentions. If we associate any communicational aspect with the appearance of the lion and the antelope's perception of that, we must negate our question. But the lion has surely no intention to be perceived. Our understanding of communication requires a *communicational act* from the agent and that implies that it has an *intention to communicate*. Therefore we suggest to define:

A ***communication*** involves at least that one agent acts upon another by the means of a linguistic action (an utterance) and the other agent reacts to this.

We could not discuss the above question without a reference to a prominent research paper, that presents a scenario where cooperation without communication is studied: Rosenschein, Genesereth and Ginsberg (RGG86) describe and analyze how agents driven by rational actions interact without any communication just on the basis of reasoning about the other agents rational decisions. Nevertheless their postulation is total knowledge about each other agent's preferences to take alternative actions. Usually that knowledge must be gained using communication. This makes that scenario a little bit strange.

⇨ In social societies, communication is the usual medium for knowledge exchange and interaction and therefore it is one of the most crucial themes for DAI.

Three conditions lead to a useful communication:

First of all agents need the ***facility to communicate*** which comprises the devices for speaking and listening. For DAI systems these facilities are usually provided by the implementation language or a message mechanism of the underlying operating system. In human society these devices are our ears and mouths, both connected to the brain[1].

The next step is the ***ability to communicate***, that is to use the communication modules (channels) to tell the other agents about the own knowledge, proposed solutions or to ask questions and to receive utterances and interpret them. So this aspect also comprises a common way of communication (language) and a mutually intelligible representation within an agent society.

---

[1] The importance of this precondition of communication and the effect of it to the evolution of intelligence is underlined by the development of speech in prehistoric times: Thought the brain of homo neandertalensis was about 100 cubic centimeters larger than that of recent humans, his larynx was located too high in the throat to produce articulated speech. His communication exhausted in miming, gestures and simple roaring and lacks the feedback via speech.

As a third point a ***communication impetus*** is presented to be a permanent yearning to tell the other agents about the own already found solutions, the current work and the plans.

Werner (Wer88a) presents several approaches to communication in form of levels with increasing complexity:

<u>no communication</u>: Agents infer other agents intentions rationally like described in (RGG86) or act without the knowledge about other agents.

<u>primitive communication</u>: The set of possible messages is restricted to a finite set of signals with a fixed meaning. Dijkstra's semaphores fall into this category as well as the simple communication via paths of pheromone used in Moyson's and Manderick's simulation of the behaviour of ants (MM88).

<u>plan passing</u>: Total plans generated by the agents are passed to other agents to help predicting the behaviour.

<u>message passing</u>: Structured messages with various possible contents are exchanged between the agents. This scheme is often successfully employed for a formal rule-governed interaction, e.g. to build up certain interaction protocols.

<u>high level communication</u>: Such a communication model must also comprise an integrated model of belief, goals, intentions and emotional states. For example the plan-based theory of speech acts by Cohen and Perrault (CP79) views speech acts as planned actions to affect the listeners beliefs and goals.

Also Werner's theory itself (Wer88b), that integrates representation, intentions and cooperation evolving out of the pragmatical effects of communication, is an approach to a high level communication. The term ***pragmatic*** denotes the effect of an utterance to the communication environment, especially the effect to the beliefs, goals, intentions... of the addressee of a message. The allocation of a goal from one agent to another as pragmatical effect of an utterance is called *goal-induction*.

➪ <u>Communication always presupposes a basis of *common structure*.</u>

Even if an utterance consists only of a signal, there must be a common interpretative meaning of it. Typical recent DAI systems communicate on the level of a usually simple message passing. This leads to an inflexible predesigned communication language. Such language systems are the result of the routinization, conventionalization and codification of the *interaction patterns*, that occur between the agents. For example in the PUP6 system (Len75) we find an interaction protocol based on the agent structure. The heterogeneous agents in this system all have a common structure and the messages always refer to specific slots of this structure.

In the already described Contract Net protocol we find a *frame like message structure* and the different message types realize the necessary communication in the protocol: Announcements, bids, award messages, interim and final reports and some special messages like termination, directed awards, refusal, node-availability. In a protocol usually most messages expect a response message.

In the next section we once more present a prominent domain of DAI research:

Distributed Vehicle Monitoring Testbed

One of the early research domains of DAI was *distributed sensing*. The **Distributed Vehicle Monitoring Testbed** (DVMT) described in (LC83) simulates a network of vehicle monitoring nodes, each connected with a few acoustic sensors distributed over the node's portion of the monitored area. The nodes shall cooperate to generate an overall correct plan of the vehicle movements (vehicle type, speed, track...) during a certain period of time, based on their local and possibly faulty sensed data.

Several researchers have used that problem domain for experiments and to develop their new ideas and concepts of different aspects of DPS. (E.g. (LC81), (DL87)) For example in the context of interaction and communication, the DVMT system was used to experimentally change the contents of communication. The possibilities were to exchange simple sensor data with adjacent nodes, abstract and condensed interpreted data, the current goals and plans for their future activity to achieve a coordinated behaviour and to improve network awareness.

The improvement of *local control decisions* was attempted to achieve with three different methods:

1.) An ***organizational structure*** that provides a long-term framework for network coordination to guide each node's local control decisions;

2.) a ***planner*** at each node that develops sequences of problem-solving activities based on the current situation; and

3.) ***meta-level communication*** about the current state of local problem solving that enables to dynamically refine the organization. (DLC87)

The first two methods already belong to the next section's topic, which indeed is strongly related to interaction and communication:

5. Cooperation, Coordination, Coherence

Cooperation contains the word operation. A cooperator is someone who works together with /or supports the operator. To cooperate means working together, pulling at the same side of the string, aiming to achieve a common goal.

➪ Cooperation usually leads to a specific kind of interaction and is achieved normally by using communication.

But the later is not necessarily the case: In (RGG86) the cooperation is based purely on the knowledge about the different agents' utilities for different action alternatives and on the agents *rationality*, which takes into account the other agents' rationality. No communication is needed provided that there is a common knowledge about the utilities, which are represented as a payoff-matrix. This approach to cooperation is very much centered around mathematical game theory, but those, who think, that this is too much abstracted away from real world problems should read (Axe84): A marvellous book around the *prisoners dilemma* presenting a variety of real world problems that contain the prisoners dilemma situation and giving much insight into the question how and why cooperation evolves even in a society of egoistic individuals.

If the achievement of a common goal increases each agent's utility, then also egoistic agents are forced to cooperate by their *self-interest*. If we enlarge our considerations to include also *antagonistic agents*, i.e. agents with different possibly even contradicting goals, we come to ***coordination***. In fact many researchers see cooperation as a special case of coordination with non-antagonistic agents. Some researchers differentiate between *positive cooperation* and *negative cooperation*, the later for example when two agents want to use a resource, that can only be used by one at a time. We would prefer to speak of coordination instead of negative cooperation in this mutual exclusion problem.

The coordination of agents requires a certain amount of extra work, indirect activity like synchronization, task alignment, prediction of other agents actions and other forms of meta reasoning, which is not directly, but only indirectly supporting the achievement of the primary goals. This work is called **articulation work** (Gas84) and could be seen as a measurement for the degree of coordination:

➪ The more articulation work can be avoided, the better is the coordination of the system.

Usually in routinized interactions with high degree of predictability and lack of conflicts, we find a good coordination, where in environments with more unexpected events and situations, that never occurred before, much articulation work is necessary. But coordination is a property of the interaction among agents.

To come to a more general viewpoint, which integrates the environment and that looks at the society of agents as a unit of behaviour, the term coherence should be used. ***Coherence*** could be evaluated along different dimensions as for example:

Solution quality: If the problem allows for different solutions: How is the systems' ability to reach a solution and what is the quality of this solution?

Efficiency: The systems overall efficiency to achieve a solution.

Clarity: The conceptional clarity of the system. This comprises the actions, the knowledge representation, the agent model, the well-structuredness and describability of the system.

Graceful degradation: How the system behaves in the presence of failures or uncertainty.

Good coordination may lead to a higher efficiency-coherence by the reduction of articulation work. For example the *focused addressing* mechanism in the Contract Net leads to a reduction of the task announcement messages. But on the other hand good local decisions do not necessarily imply better global behaviour as is shown in (KHH88), where a more intelligent local behaviour has negative effects for the systems coherency.

⇨ The problem is to achieve coherence and cooperation without a centralized control or centralized viewpoint.

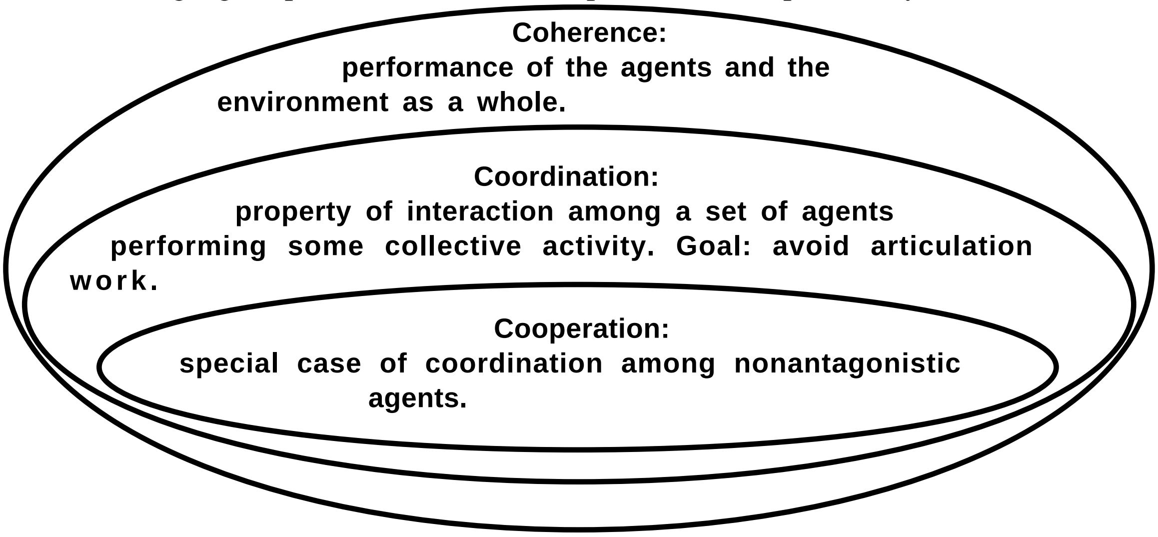
Lesser and Corkill (LC87) present the following as necessary conditions for a solution-coherent behaviour:

coverage: Each necessary portion of the problem must be included in at least one agents activity.

connectivity: Interaction between the nodes necessary to develop and integrate a solution.

capability: coverage and connectivity must be achievable within the resource limitations of the network.

The following figure presents the relationship between the previously introduced terms:



One can think of several strategies to improve a system's coherence, but their real effect will strongly depend on the actual considered system. We will sketch some examples of more general strategies:

Strategies to Improve Coherence

Increase the ***localization*** by the *specialization of the nodes*, thus giving more room to a focused addressing. Another way to increase the localization could be to *reduce the local dependencies* of nodes e.g. by introduction of more resources or a better task decomposition.

To increase the nodes ***contextual awareness*** may have the effect of avoiding articulation work. Confer (Mül91) for a discussion of this strategy in a sample-scenario. If an agent has knowledge about other agents plans, goals and activities and an enriched knowledge of the problem domain, it will be able to predict other agents actions more reliable. Usually communication between agents will be more expensive than maintaining a local knowledge base.

A more sophisticated ***management of communication*** may also improve coherence and cooperation. Communication is motivated by sharing time varying information about the perceived external world as well as information about the agents internal state. A communicated message could be rated according to its

<u>relevance</u>,	i.e. its consistency with the global solution, according its
<u>timeliness</u>,	i.e. its influence on the receivers current activity or its
<u>completeness</u>,	that is the fraction of a complete solution, that the message contains.

Such attributes could help making better communication decisions. For example messages with a small timeliness should not be send. Messages, that are consistent with the smallest number of current hypotheses, may have a high discriminating effect, that is, they may rule out many hypotheses.

Minimizing interdependencies between agents could also lead to more coherency. The idea is to relax the tightness of coupling of tasks. Tasks are coupled if the input of the one task depends on the output of another task and are tightly coupled when state changes in one task immediately effect the state of the other task. To reduce the tightness of coupling, **slack resources** are introduced. Two interpretations of slack can be found in distributed systems:

First	the replication of a task on an alternate processor if one processor fails.
Second	the replacement of procedure calls by message queues; thus reducing the tight coupling of agents by the introduction of buffers.

Coping with uncertainty is another problem relevant for coherence.

Uncertainty is the difference of the available information and the information necessary to achieve the best decision.

Uncertainty could be subdivided into *information uncertainty*, *decision uncertainty*, *environmental uncertainty* and *behavioral uncertainty*. Behavioral uncertainty for example, that occurs when agents do not deliver on their commitments, has a great impact on the agents coordination. Decision uncertainty, that is uncertainty due to the lack of knowledge about the outcomes of decisions, could have a great effect to the solution-coherence.(BG88)

Central for the "***scientific community metaphor***" is the *competition* between different in parallel developed viewpoints. In Hewitt's terminology these viewpoints are integrated by "*due processes*". Corkill and Lesser speak of a **functionally accurate**, **cooperative behaviour** (LC81), that produces a solution by combining evidences of competing hypotheses. Mutually inconsistent hypotheses die out, while consistent hypotheses grow together and magnify their evidences.

One prominent domain of research in strategies of cooperation is the air traffic control scenario:

Air Traffic Control

The **Air Traffic Control** (**ATC**) problem (CMS83) (SCHTW81) involves all important aspects of DPS from sensing situational data over planning to cooperation and resolving disparities:

Aircraft enter a rectangular airspace around an airport either at one of 10 infixes of the border or from the airport. The goal for each aircraft is to traverse the airspace to a destination, which again is a boundary outfix or the airport. Each agent (=aircraft) has only a limited sensory horizon, hence its knowledge of the world is never complete and it must continually gather information as it moves through the airspace. Information may be accumulated either by *sensing* or *communication*. Agents are allowed to communicate over a limited band-width channel to other aircraft for purposes of exchanging information and instructions.

ATC is a *group problem* because the agents may help each other to gather information, but also because the goals of one agent may interact with those of another agent. *Goal interactions* come in form of ***shared conflicts***:

A conflict between two or more agents arises when, according to their current plans, the two will violate minimum separation requirements at some point in the future. When shared conflicts arise, agents must negotiate to solve them. In a crowded airspace, such goal conflicts can get particularly complex, and involve several aircraft, thus necessitating a high degree of group cooperation.

In (CMS83) four organizational policies are analyzed. The first three use ***task centralization***, which means, that one conflict partner is selected to evaluate the conflict conditions, replan and act according to the new plan. The other conflict partners send their intended routes to the first one, but stay passive i.e. do not change their plans. The differences of these three methods lay in an increased knowledge that the agents use to select the active agent. The fourth method, ***task sharing***, distributes the replanning task and the actor that performs the new plan to different agents and requires therefore a more intensive communication.

Another important aspect of coherence and coordination is planning:

Planning

Planning is a prominent research area in classical AI and could not be discussed in the frame of this text, but everyone who deals with rational agents should have a good knowledge about planning. DAI terminology splits planning into

multi-agent planning: A single planner generates plans for multiple agents, - and

distributed planning: Not only plan execution, but also plan generation is distributed among multiple agents.[2]

➪ The crucial point is to enrich the *classical planning* with synchronization mechanisms.

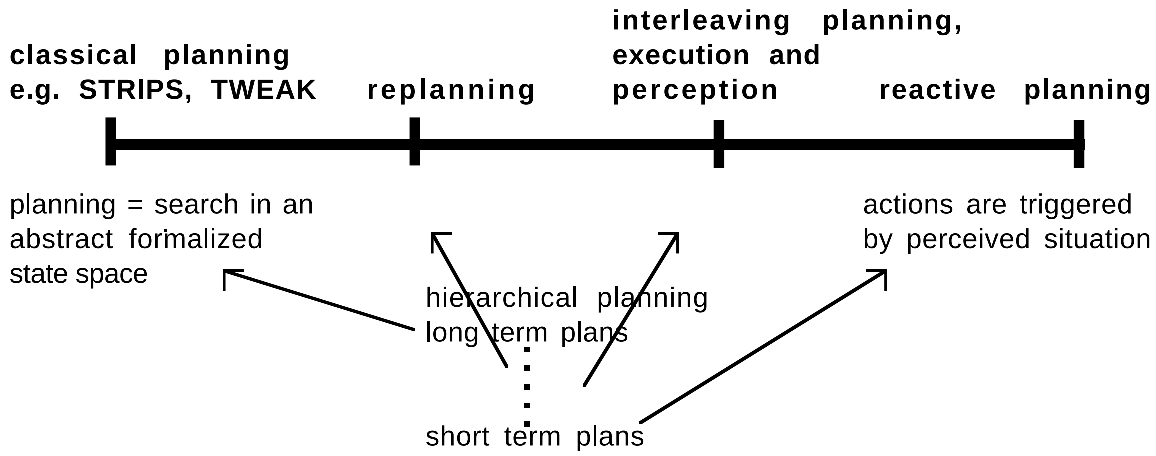
Wether it is sufficient to insert synchronization operations after the planning into single agent plans, or whether it is necessary to reason about the dependencies as an integral part of the plan from the beginning, is a question of the specialities of the domain. Each dynamic changing environment and every world, where a formal detailed specification of the preconditions and effects of a predefined set of operators is not possible, in other words, every real world problem requires a form of ***replanning*** in response to the changing problem situation. Interleaving of planning, execution and perception is a further step to a reconciliation of the extreme positions of classical planning and the *reactive behaviour approach* , where the actions of an agent are triggered by perceived events in the environment. Another approach towards a more adaptive strategy would be a form of ***hierarchical planning*** over many levels like we can find it in our own human problem solving behaviour:

> Planning long term goals in abstract plan steps is often more projective and may involve sketchy plans, uncertainty and relaxed constraints. Resolving abstract plan steps more and more leads to short term plans, that require more behavioral and situation adapted aspects for their execution.

---

[2] I am not very happy with this terminology, because the term multi-agent system usually refers to a society of autonomous agents each with own planning capability. Thus multi-agent systems use distributed planning, while distributed problem solving uses multi-agent planning. That's confusing.

The following figure represents the previous mentioned approaches as points in a continuum between the two extreme positions of classical planning and reactive planning. Hierarchical planning attempts to comprise more than one dimension of the continuum:



6. Organizations

The most important technique to improve a systems coordination and coherence and to deal with increasing complexity is to install a kind of *organizational structure* in the system. Therefore, a whole section is devoted to this theme though it is conceptual one methodology, that belongs to the previous section. Corkill, Lesser and Durfee (DLC87) view organizations as a long term strategic load balancing technique to improve coherence. The questions are:

- ➡ How can an organization be modelled?
- ➡ What different types of organizations exist?
- ➡ How can organizational knowledge be represented?
- ➡ How reasonable is it to transfer ideas and concepts from human organizations to societies of computer-based agents?

An ***organization*** is a complex framework of communication paths, expectations and patterns of behaviour.

Usually a certain set of expectations, behaviour patterns or attributes are associated with a ***role***. A role can be taken by different agents; for example in the Contract Net protocol the role of a manager can be taken by each agent. A role *commits* an agent to a specific behaviour. From this viewpoint an organization results from restrictions on a wide range of possible behaviour patterns, therefore a role has to specify which attributes, which methods and messages are enabled or disabled. Each concept to model an organization has to specify in which way the roles are represented.

Organizational Models

One possibility is to specify roles by locally interpreted ***rules of behaviour***. This matches with the intentions of the designers of the PUP6 system (Len75), where the rules of behaviour are derived from the common agent structure. The BEINGs system represents a “flat” organization with one moderator agent as a specialist for conflict resolution. Other agents are specialists for specific domain tasks. The model for BEINGs was the paradigm of communicating experts, that develop a solution in group-work (<u>scientific community metaphor</u>). Such an organization has to rely on the fact that the local rules of behaviour result in a consistent and coherent global behaviour.

Werner (Wer88b) does not use the term organization, but outlines his view of ***social roles***, that could be used to model various types of organizations within an agent society. In his model, the *representational state* of an agent consists of an *information state*, an *intentional state* and an *evaluative state*. A social role in this context is now an abstract representational state, which is also composed out of these three components. An agent takes a role by integrating this abstract representational state into his own representational state.

In the model of a ***centralized hierarchical organization***, we find a cascade of abstraction levels. The motivation for this is to achieve greater control and a more global viewpoint. This organizational form is usually encoded in the structures, attributes and actions of the heterogeneous agents of the different levels, and thus not very flexible. Typical for a hierarchical structure is, that decision making data flow upwards the cascade, while control flows downwards. A special aspect of this model is to enrich the organization with an *authority structure*. A lower node must accept goals, results or decisions of a higher node. But this reduction of coordination work by the preference of the more authorized node does not necessarily imply an increasing solution coherence. This is only achieved, if the more authorized node has enough information to make the “right” decision.

Inspired by the mechanisms of supply and demand in economical markets, researchers have tried to design DAI systems as ***competitive market-like organizations***.The competitive bidding mechanism of the Contract Net protocol is one example of this. Research in this direction should investigate the mechanisms, that produce coherent global system performance as emergent effect of many local economical decisions. The central dogma of free enterprise economy, which states, that the public welfare will be maximized if every atomic economical unit maximizes its own welfare, sounds very promising to the advocates of the reactive behaviour paradigm. If simple local rules for the main problems of DAI, which work this way, could be found, the reactive behaviour approach would be very successful. But as we know from economics, things are not as simple as theorists model them.

We've never heard of an organizational model in DAI, that is oriented on a *metabolic paradigm*. The regulation of enzymatic activity (,which catalyzes the metabolism,) by genes could be another model for the DAI reactive behaviour approach and also another example how to learn from other branches of science.

A special form of organizational model is realized by an early and very prominent DAI system; the blackboard architecture of the speech understanding system HEARSAY-II:

Blackboard Architecture

In a **blackboard architecture**, the expert knowledge is encoded in several ***knowledge sources*** (KS), each running as an own process. The KSs share a common data-structure, called the ***blackboard*** (BB), which is used to develop the solution and as a communication medium for the KSs. The symbolic structures on the BB are called ***hypotheses***. Each KS is an expert in some area and may find a hypothesis it can work on, solve that hypothesis, create new hypotheses undermine some of them or modify other existing hypotheses. The KSs respond opportunistically to changes on the BB.

The most prominent and archetypical BB-architectured system is the HEARSAY-II speech understanding system (EHLR80). The hypotheses are organized in a hierarchy of abstraction, which is reflected by the *separation of the BB into different levels*. In HEARSAY-II for example we find levels for segments, syllables, words, word-sequences and phrases. A KS consists of a pattern and an action. An activation of the KS is created, whenever a hypothesis on the BB matches the pattern. The ideology of the BB-architecture is to use the "hypothesize-and-test-paradigm" and thus the KSs usually create several hypotheses, that support different interpretations. An *opportunistic and asynchronous style of problem solving* is achieved with a scheduler, that rates all KS activations according to their impact on the current state of problem solving. Thus a priority queue is maintained from where the first activation is selected for running. The rating is based on such attributes like the *credibility* of hypotheses, the *scope* of some information, which refers to the level in the hierarchy, or the *diagnosticity*, that says how much competing hypotheses can be ruled out as an impact of the activation. The scheduler uses meta-information from a special *focus of control data-base*.

The BB-architecture of HEARSAY-II, though itself not distributed, had have a large impact on the early DAI research and several distributed systems had modelled their single nodes like HEARSAY-II systems (LE80) (Gre82) (Gre87).

Organization Types

The question about different types of organizations is closely entangled with the organizational model. Fox in (Fox81) presents various organization types in economy and analyzes the evolution of these types along the dimensions of complexity and uncertainty. *Uncertainty* - to repeat this - is the difference between available information and information necessary to make the best decision. *Complexity* is seen by Fox as excessive demand on *rationality*. The *bounded rationality* of humans is the prime factor in the evolution of organizations. A transition to a more complex organizational type is necessary when a certain limit of capacity to deal with rationality is exceeded. Fox traces along the following organizational types:

single person

group : Coordination through mutually agreed decisions, all members have all available information.

two level hierarchy : A decisionmaker with complete information and authority guides the other nodes.

multiple level hierarchy : Many levels of management over the bottom layer, each is a filter on information and decision.

multidivisional hierarchy : Division of the organization along the different products of a company, added by an elite staff group for strategic control decisions. Another possible division could go along the production steps of the different products (-> functional division).

price system : The introduction of a price as the result of a negotiated contract eliminates all other forms of control. Control is exerted via the price. Contracting opens the possibility of a dynamic creation of the system architecture.

collective organizations : As a result of an elaborated price system, a set of organizations develops, each of which for itself can be of one of the previous organizational types.

full market : As the number of organizations, that produce the same or similar products increases, the increasing competition installs a full market, where the price is fixed by demand and supply.

Representation of Organizational Knowledge

Organizational knowledge comprises some *general principles* of organizations, knowledge about *specific methods* used in the organization, *expectations about the actions* of other members of the organization and possibly a structural view of the *relationships* among the agents. The representation of organizational knowledge is very much related to the implementation of the interaction. Different aspects of an organization will be represented in different ways.

The *structural aspect* for example could be represented by a *graph*, whose nodes may represent agents, tasks or products connected by relations of authority, communication, control or information flow.

Functional roles could then be represented by lists of *capabilities* or constraints on agents activities.

Planning oriented aspects will be handled convenient if we represent an organization as a *collection of tasks or actions*.

The *interactions* within the organization are represented appropriately by *collections of expectations, commitments and behavioral defaults*. This will require an explicit or implicit model of other agents as will be described in the next section.

Finally the *general principles* of an organization will usually be represented implicitly *encoded in the agents internal structures and capabilities* and in the system architecture.

The use of a particular architecture, language or shell determines the organizational form of the system. The Contract Net protocol for example forces a strict virtual hierarchy with no lateral communication, but as we have seen allows a decentralized task decomposition and a certain flexibility to the systems load. A blackboard shell (see (Hay85) for a good description) will force the typical blackboard architecture, with knowledge source nodes, that have the fixed roles of experts and share a common datastructure, the blackboard, where the solution is developed.

Transfer of Concepts from Human Society Organizations?

The question how reasonable it is to transfer ideas and concepts from human society organizations to DAI agents and even any other distributed systems, can only be answered by the success of such approaches like the one of Fox and their implementational results. Of course just now it is not rational to rebuilt human society organizations with artificial agents, because the bounded rationality of humans is still of another quantity and quality than that of artificial agents. But the transfer of concepts and ideas seems to be the only rational way, because from where else should we get our concepts, if we consider the overall goal of DAI, which indeed is to approach the human society paradigm?

7. Modelling other Agents

Again we will try to structure this section along some relevant questions about agent modelling:

- ➡ What is a <u>model</u> of an agent?
- ➡ <u>Why</u> do agents need models of other agents?
- ➡ <u>What knowledge</u> about other agents must be represented in such a model?
- ➡ <u>How</u> is this knowledge <u>organized</u>?

Agent Model

By “agent model” we do not mean the modular structure of the agent, that the designer of a DAI system tries to implement. Thas is a problem of knowledge representation and agent architecture. In this section we mean the following:

An ***agent model*** comprises everything that allows an agent to have some expectation about another agent's structure or behaviour.

This could be something very simple, for example we stated earlier, that a certain amount of common structure at least concerning the interpretation of messages is a necessary precondition for each communication. Just the reliance, that the other agent will interpret messages in a certain way, is a kind of *implicit encoded model* about the receivers interpretation algorithms. I as the author of this text have the expectation, that you as the reader of this text have the ability to read and understand it, or another example: a typewriter could be viewed to have a user model encoded in the shape and structure of its keys.

Every meaningful interaction presupposes at least an implicit knowledge of each other. Even in the scenario presented in (RGG86), where no communication between the agents occurs, we can find a rudimentary form of a model: Both agents understand the representation of their utility in the payoff-matrix and both assume, that the other agent bases its decisions on its rationality. The term **network awareness** (DLC87) denotes the knowledge, that a node uses to cooperate with other nodes.

⇨ The knowledge, that the agents enable to communicate according to a protocol as e.g. the knowledge of other agents addresses, their capabilities, already assigned contracts with them and so on builds the agent's network awareness.

The agents' behaviour according to the protocol and the expectation, that other agents will perform the protocol as well, implicitly encodes their model of other agents.

More sophisticated and *explicitly represented models* of other agents will enable an agent to take into account another agents knowledge, actions, plans, goals or beliefs. In existing DAI systems these *higher level models* are based on heuristics or abstracted views of the other agents. Precise propositional models of other agents are merely used for theoretical accounts. Such models must contain adequate concepts of knowledge, belief and intention. Two major approaches to model knowledge and belief can be distinguished in classical logics (cf. (GN87)):

The ***sentential logic*** of belief attaches to every agent a database - the agents beliefs, - and a set of inference rules. The reasoning capabilities of an agent correspond to its inference rules and might be restricted and incomplete. To prove, that an agent has a certain belief, one has to simulate the agents internal reasoning by applying its inference rules to its database (if the belief is not already in the database).

The other approach uses a ***possible world logic***: A graph of possible worlds, each with its own interpretation of formulas, connected via an accessibility relation, represents the agent's situation. An agent knows a formula, if it is true in all worlds, that it could reach from its actual world via the accessibility relation. Certain properties of the accessibility relation reflect certain qualities of knowledge. For example the knowledge axiom, which states that an agent should not know something which is false in its actual world, can be modelled by a reflexive accessibility relation.

A kind of *benchmark* for the power of a logic of knowledge and belief and a good presentation for the subtleties of beliefs about others knowledge and beliefs (<u>nested beliefs</u>) is the **wise-man puzzle** proposed by McCarthy in (MSHI78):

> A king invites three wise men and paints a colored point on each mans forehead such that everyone can see the color of the others' points, but cannot see his own point. The king states, that at least one of them has a white point and asked each about his color. After the first two men have stated, that they did not know their color, the third man should be able to infer, that his point is white, by considering the other men's reasoning.

This puzzle involves nested believes and can be extended to more than three men, to increase the degree of nesting. A sentential logic approach, that is based on the *amalgamation of object language and meta language*, for solving the wise-man puzzle is presented in (KK90).

Sociologists have claimed, that internal models of oneself and other agents are a primary factor for the creation of an organized society. The demand to (partially) model oneself follows naturally from the demand to model others. Objectifying oneself by applying the same modelling concepts used for other agents to oneself seems to be a first step to consciousness, to become aware of oneself. Approaches to this goal could be found under the catchwords ***introspection***, ***reflection*** or ***autoepistemic knowledge*** in explicit and logic-based models. The goal is to define an adequate metalevel-language, which is capable to speak about the logic and the inferences performed in the ground-level logic. Confer (Dav90) for an overview of logic based meta-knowledge.

Motivation for Agent Models

One motivation, why agents should have models of others has already been presented: The commonality for the interpretation of communication is a *necessary precondition for each meaningful interaction* [3]

---

[3] The ascription of a user model to a typewriter is a little bid a question of the point of view and also in some MA-systems like e.g. in (MM88), where the behavior of ants is simulated, it is hard to argue for even an implicit model of other agents used for interaction.

Higher level models are useful because of their ***predictive power***. An agent may be able to predict events that it cannot sense directly like the changes of beliefs and plans in another agent's database. Messages between agents to inform each other about such changes may become unnecessary, which will decrease the articulation work and increase the coherence.

Models will also help to improve the efficiency by a *reduction of overhead*. In the Contract Net protocol the mechanism of focused addressing reduces the number of sent task announcements; but focused addressing is only possible, when the node knows about special capabilities or resources of other nodes, which is a simple form of model of these other nodes.

Furthermore, an agent model may be useful in *ranking the reliability, usefulness or timeliness of data*.

One problem with higher level models is, that because it is not useful to simulate the whole other agent in the model again, the question arises:

Which part of the agent is relevant for the model?

The model should enable to predict the other agents default behaviour or routinized actions, but not unusual and seldom occurring cases of its behaviour. These cases should be coordinated by extra negotiations. Another problem concerns the timeliness of predictive information:

When should predictive information be deleted?

Answers to these questions heavily depend on the concrete scenario to be modelled.

Contents of Agent Models

By *communicational knowledge*, we mean knowledge, that is necessary to communicate with other agents. It can be subdivided into a *formal part* and a *contents-part*.

The formal part comprises knowledge about ***channels***, ***addresses***, ***languages***, ***protocols***, ***message***, ***formats*** ... and all knowledge that is necessary to perform communicational acts generally.

The contents-part comprises all kinds of knowledge, that is necessary to decide, what information should be communicated in the actual situation.

Other agent's ***capabilities*** and their therefore demanded ***resources*** are relevant for task allocation considerations, feasibility analysis or performance assessment. Allocation decisions could depend on a pre-calculation of the temporal or resource consuming behaviour of the agents.

Knowledge about other agents' ***responsibilities*** could be used to install organizational structures.

Knowledge about an agents ***processing state*** may be useful for a supervisor agent to gain a picture of the overall solution progress.

Actions, ***plans***, ***goals***, ***believes*** and ***intentions*** of other agents are used to reason about the effects of actions. Joint actions require several agents to anticipate an action. An explanation of actions is only possible, if a notion of an agents belief-system and its intentions is available. Certain intentions and behaviour patterns can be abstracted into a role and reversely an agent may infer another agents actions by the knowledge of its role.

Organization of Agent Models

The organization of knowledge, that comprises the modelling of other agents, depends on the level of this model and very much on the specialities of the system. Implicit models are organized in form of ***protocols***, ***interpretation algorithms*** and ***interaction expectations***. But even such implicit models often involve some necessary explicit information. *Addresses* and *capabilities* of other agents for example.

The MACE system (described in GBH86) for example provides an associative database, that allows an agent to retrieve knowledge about others via agents, via tasks, goals, plans, skills or organizational roles. If the agent wants to know what capabilities a specific agent has, it can use the agents name as a search key, but if it is looking for a contractor with a special capability, a query with that capability as search key will provide all relevant agents for its demand.

Propositional models, though not mature for practical systems, will represent models of other agents and introspective models as logical *formulas* and *axioms*. Specialized *reasoning rules* tailored for example to introspective reasoning or encoding some heuristics could be provided.

8. Disparities and Conflict Resolution

The reason of conflicts is laid in the necessity to represent at least parts of the world.

⇨ This *objectivation* involves *abstraction* and introduces *incompleteness* and thus *uncertainty* about the real world objects, which includes the other agents and their internal states.

If all agents would represent the same part of a problem domain, use the same representation and draw the same reasoning steps based on the same interpretation, then all agents would do identically the same. So there must be some ***disparities*** between the agents for a useful work, but these disparities also may cause conflicts between the agents. The problem is to manage these disparities of knowledge and representation, i.e. to resolve the occurring conflicts by appropriate methods.

Common representations and standard communication protocols are designed into the most existing DAI systems, but this does not solve the whole problem. The following *forms of interagent disparities* in agents' knowledge bases can occur:

- <u>incomplete beliefs</u>: Different agents' knowledge bases contain different beliefs. In many DAI systems, each node only has a partial view of the scene, like in the DVMT and the ATC system for example. Incomplete beliefs of this sort are normal and the solution is found by aligning the different knowledge bases by an exchange of knowledge of several abstraction levels towards a globally complete and consistent solution.
- <u>logical inconsistence</u>: One step worse than incomplete beliefs is the contradiction of the beliefs of one agent with the beliefs of another agent. This cannot be cured by pure exchange of knowledge, but one agent has to persuade the other agent by a prefixed priority, by argumentation and justifications of its beliefs or by invoking a higher level conflict resolution mechanism.
- <u>confidence inconsistence</u>: Similar knowledge is represented in different agents at different confidence levels.
- <u>incompatibility</u>: Different agents use different forms of representation for their knowledge. Usually the system designers will install a common or easily translatable knowledge representation in the agents. But if we want to build a multi-agent system from several existing stand alone systems, expert systems for example like in the ARCHON-project (cf. (Wit90)), we have to face this problem.

➪ <u>It has been argued, that in all large scale distributed systems we'll find ***inherent inconsistencies*** and that therefore it is sufficient to keep the nodes' knowledge bases locally consistent.</u>

The overall goal - to say this clear - cannot be to design a DAI system with no disparities and conflicts at all. On the contrary,

➪ <u>many researchers have pointed out the healthiness of duplicated, variant, different and even conflicting activities.</u>

To reconcile the disparities and conflicts is often the kernel of the work of finding a solution. In the ATC system alternative solution plans generated by the different nodes are merged together to produce an overall plan. Also in the DVMT **partial global plans** are merged together and in the **functional accurate/cooperative behaviour** proposed in (LC81), the process of finding a solution depends also heavily on the variety of partial solutions and the dying out of mutually inconsistent solution parts.

Using the scientific community as a metaphor for the organization of problem solvers in a MAS, we could draw the conclusion, that

➪ <u>progress evolves out of a set of conflicting explanations of phenomena.</u>

A special kind of natural-selection mechanism of theories evaluates the most explanative theory. Dawkins (Daw76) shows surprisingly close analogies between the evolution of genes and memes as he calls the "units of replication" in our brains.[4]

Kornfeld and Hewitt (KH81) try to formalize the scientific community metaphor by suggesting three types of agents, ***proposers***, ***proponents*** and ***sceptics***, which exchange two types of messages: ***Conjectures***, that account for observations and ***refutations***, that undermine conjectures.

An appropriate handling of agent disparities requires the recognition and representation of disparities and therefore the representation of other agents beliefs, plans and goals. The last subsection already presented the problems of modelling other agents. We want to complete those considerations only by two more hints to the literature:

Bruce and Newman (BN78) tackle the problem of common beliefs by the introduction of a **mutual belief space** mechanism. They divide the beliefs of each agent into two parts, those, that are held commonly (mutual belief space) and those that are incomplete and disparate for different agents. This method avoids an explicit representation of nested beliefs, because the mutual belief space is shared by all agents and all agents know this. The mutual belief space is also useful to represent all active beliefs during a certain cooperative interactive phase of processing to improve cooperation without extensive transmission of messages.

Halpern and Moses (HM84) analyze the problem of common knowledge from a theoretical viewpoint and proof, that common knowledge is impossible to achieve without the *guarantee of message arrival with a finite time delay.*

In principle a method can be based on *power*, which means that one agent dominates another, or on *mutual agreed conflict-resolution strategies*, which can be based on conventions about priorities or on specific mediation procedures. In the following we will present some possible conflict and disparity resolution strategies according to (BG88):

Conflict Resolution Strategies

Abstraction of common frameworks: Disparities in representation are often caused by different levels of description. Adding details by *elaboration* or

[4]This reference deals with biological evolution, not with DAI. Nevertheless, besides the cited analogy, Dawkins also presents some very interesting results of applying the framework of mathematical game theory to the evolution of behavior patterns in animal populations.

removing them by *abstraction* could be one way to realign disparate representations. Translation of different representations into a common one or a common language could also make different representations compatible. The common framework as the basis for this process could be realized in form of a homogeneous agent structure like in (Len75) or as an abstract protocol like the Contract Net protocol.

Achievement of common knowledge by integration: If conflicts are based on incomplete knowledge, agents could complete their knowledge by *communication* with other agents, that have this knowledge or by using their own *reasoning* or *perception* capabilities to gain this knowledge. This presumes common compatible representations and mechanisms for adding and incorporating the new knowledge, which may include consistency-checking.

Assumption surfacing: This strategy assumes, that agents keep knowledge about the supporting assumptions of their beliefs, e.g. by using a truth maintenance system. If disparities between the agents' knowledge occur, they are reconciled by *backtracking through the assumptions* to discover the root of the disparities.

Authority and knowledge mediation: Higher level knowledge is accompanied with authority to resolve conflicts (by mediation). This can be *centralized* as in the HEARSAY-II system, where it is encoded in the heuristics of the blackboard scheduler, or can be *decentralized* as in the ATC system, where each node rates its goals according to conventionalized priority rules. Here these rules encode the higher level knowledge.

Constraint relaxation: If the problem is formulated with constraints, a relaxation of constraints may be helpful for a reconciliation of conflicts. A priority of constraints is required such that weak constraints can be relaxed first.

Evidential reasoning: Evidential reasoning in the context of justifying conclusions of knowledge-based systems is a well known concept in AI. It could also be tailored to DAI. Lesser's and Corkill's functionally accurate/cooperative processing (LC81) uses evidences to strengthen the most favorable hypotheses and to converge to a consistent solution, while less favorable and mutually conflicting hypotheses die out.

Goal priorities: Associating priorities to goals allows reasoning about the effect of choices in resolving disparities. Which disparities are important enough to be solved? This mechanism can be found in the DVMT and in blackboard systems to make decisions about the scheduling of knowledge sources. In typical rule-based systems this is called *conflict resolution mechanism*.

Negotiation: Is used to exchange information and to resolve conflicts. A negotiation is held in a *context* about a set of (disparate) *goals*, involves some *information* and *knowledge* and is executed according to some rules encoded in a *language* or a *protocol.* Ideally a negotiation protocol should provide the capability to shift the level from the object-level to some meta-level topic like the agent's interaction and their roles within the negotiation. All the other presented strategies for conflict resolution may be applied within the negotiation. *Flexibility* is one of the major requirements, that a negotiation framework must meet. The Contract Net protocol for example as a framework for negotiated task allocation is relatively weak, because it provides only a very fixed frame for the negotiation. No mechanisms of *compromising*, of *judging credibilities*, *weighing priorities* or persuasion by *argumentation* are supported.

Alternative representations: Modifying or changing the representation can help to "see the problem in a different way", which may help reconciling conflicts. This can be done by adding new knowledge, by translating into another representational scheme or even by simply changing the sequence of the goals to explore.

Standardization: Analogies and similarities in the occurring conflicts and the mechanisms, that successfully resolve them, may lead to a standardization of these mechanisms. This could be done by installing *default behaviors* for specific problem classes or *adaptively refining numerical values* which may represent evidences or priorities. Standardization can be designed into the system or may be learned by the system over time if appropriate learning capabilities are provided.

9. Conclusions

One of our goals was to develop a *morphological box*, that captures all possible forms and combinations of MAS. A morphological box describes a thing (e.g. a chronometer) along several attribute-dimensions (e.g. display-type, propulsion, size ...), such that each attribute can take several values (e.g. propulsion by battery, by a spring, by weights ...). If this description is complete, the combination of all values of the attributes comprise all possible types of the thing. Also conflictual combinations (e.g. propulsion with weight in a wrist watch sized chronometer) are captured in the box and must be ruled out by common sense considerations. The following figure presents an attempt of a morphological box for DAI-systems. It is not complete and in some way more a collection of catchwords, because the technique of the morphological box fits better to technical devices than to research areas. Nevertheless, it summarizes our work and may help to gain an overview of the current DAI research area:

attributes	values				
system type	Multiagent-system	Distributed Problem Solving			
Knowledge Representation	behavior-based	logic-based			
decomposition criteria	space	time	interest area	predefined by the problem	with redundancy
allocation scheme	Contract-Net like	according to the organizational role	distributed planning	multiagent planning	
recomposition mechanism	negotiation	automatical	functionally accurate/ cooperativ behavior		
communication level	no communication	primitive communication	plan passing	message passing	high level communication
steps to meaningful communication	facility to communicate	ability to communicate	communication impetus		
coherence	solution quality	efficiency	clarity	graceful degradation	
organizational paradigm	restrictions of behavior patterns	rules of behavior	scientific community	social roles	
	blackboard system	centralized hierarchical org.	competitive marketlike org.		
organizational type	single person	group	two level hierarchy	multiple level hierarchy	multidivisional hierarchy
	price system	collective organizations	full market		
agent model	implicit	network awareness	sentential logic	possible world logic	autoepistemic knowledge
interagent disparities	incomplete beliefs	logical inconsistence	confidence inconsistence	incompatibility	
conflict resolution based on	dominance	mutual agreed priorities	negotiation protocol		

Task of this report was to gain an overview of the comparatively new research area of DAI. Till today there is no homogeneous textbook, but there are a few books, which are collections and presentations of prominent and influencing *research papers* (e.g. (BG88) (Huh87) (GH88)). Another source of material are *proceedings* of workshops and conferences about that topic (e.g. (DM90) (DM91)). The problem with the most research papers is, that they present isolated approaches to specific problems. So the current DAI research area appears to be like an archipelago: It is necessary to make the sea level sink to see a coherent landscape of DAI.

10. References

Axe84 Axelrod, R.: The Evolution of Cooperation, Basic Books , Inc.,New York 1984

BG88 Bond, A., Gasser, L.: Readings in Distributed AI, Morgan Kaufmann, Los Angeles, 1988

BN78 Bruce, B., Newman, D.: Interacting Plans, Cognitive science, 2(3):195-233, 1978, also in (BG88)

Bro86 Brooks, R.A.: A Robust Layered Control System For A Mobile Robot, IEEE Journal of Robotics and Automation, Vol.Ra-2,No.1, March 1986

Bro91 Brooks, R.A.: Intelligence without representation, Artificial Intelligence 47, p.139-159, 1991

BS85 Brachman, R.J., Schmolze, J.G.: An overview of the KL-ONE knowledge representation system. Cognitive Science 9(2), 171-216, 1985

Cha87 Chapman, D : Planning for Conjunctive Goals, in Artificial Intelligence 32, 333-377, 1987

CL90 Cohen, P.R., Levesque, H.J.: Intention Is Choice with Commitment, Artificial Intelligence, Vol. 42, Numbers2-3, p. 213-261, March 1990

CMS83 Cammarata, S., McArthur, D.,Steeb, R.: Strategies of Cooperation in Distributed Problem Solving, Proc. of the IJCAI83, p.767-770, 1983,.also in (BG88)

Coe88 Coelho, H.: Interaction Among Intelligent Agents, ECAI88, 717-718, 1988

CP79 Cohen, P.R., Perrault, C.R.: Elements of a Plan-Based Theory of Speech Acts, Cognitive Science, 3(3):177-212, 1979

CSW88 Connah, D., Shiels, M., Wavish, P.: Towards Artificial Agents that can Cooperate, Technical Note 2643, Philips Research Labs, England, 1988

Dav90 Davis, E.:Representations of Commonsense Knowledge, Morgan Kaufmann, San Mateo, CA., 1990

Daw76 Dawkins, R.: The selfish Gene, Oxford University Press, Walton Street, Oxford, 1976

DL87 Durfee, E.H., Lesser, V.R.: Using Partial Global Plans to Coordinate Distributed Problem Solvers, Proc. of the IJCAI87, p 875-883, 1987, also in (BG88)

DLC87 Durfee, E.H., Lesser, V.R., Corkill, D.D.: Coherent Cooperation Among Communicating Problem Solvers, IEEE Transactions on Computers, C-36:1275-1291, 1987, also in (BG88)

DM90 Demazeau,Y., Muller, J.-P.: Decentralized Artificial Intelligence, Proc of the first workshop on Modelling Autonomous Agents in a Multi-Agent World, Elsevier Sc. Pub/North Holland, 1990

DM91 Demazeau,Y., Muller, J.-P.: Decentralized Artificial Intelligence, Proc of the second workshop on Modelling Autonomous Agents in a Multi-Agent World, Elsevier Sc. Pub/North Holland, 1991

DS83 Davis, R., Smith, R.G.: Negotiation as a Metaphor for Distributed Problem Solving, Artificial Intellegence 20(1):63-109, 1983

EHLR80 Erman L.D., Hayes-Roth, F.A., Lesser, V.R., Reddy, D.R.: The HEARSAY-II speech understanding system: Integrating Knowledge to Resolve Uncertainty, ACM Computing Surveys, Vol. 12, pp. 213-253, 1980

Eis91 Eisinger, N.: An Approach to Multi-Agent Problem-Solving, Technical Report of the ECRC GmbH, Arabellastr.17, 8000 München 81, ECRC-91-8, 1991

FN71 R.E.Fikes and N.J.Nilsson, STRIPS: A new approach to the application of theorem proving to problem solving, Artificial Intelligence 2, 1971

Fox81 Mark S. Fox. An Organizational View of Distributed Systems. IEEE Transactions on Systems, Man and Cybernetics, SMC-11:70-80, 1981

Gas84 Gasser, L.: The Social Dynamics of Routine Computer Use in Complex Organizations, PhD thesis, Dept. of Computer Science, University of California, Irvine, CA Feb. 1984

GH88 Gasser,L.,Huhns,M.N.: Distributed Artificial Intelligence, Vol.II, Pitman/ Morgan Kaufmann Publ., San Mateo,CA, 1988

GN87 Genesereth, M.R., Nilsson, N.J.: Logical Foundations of Artificial Intelligence, Morgan Kaufmann Publishers, Inc., Los Altos, CA, 1987

Gre82 Green, P.E.: Distributed Acoustic Surveillance and Tracking, Proc. Distr. Sensor Networks Workshop, pp. 117-141, M.I.T. Lincoln Laboratory, Lexington, MA, Jan. 6, 1982

Gre87 Green, P.E.: AF: A Framework for Real-Time Distributed Cooperative Problem Solving, in (Huh87), pp. 153-175

Hay85 Hayes-Roth, Barbara: A Blackboard Architecture for Control. Artificial Intelligence Journal, 26:251-321, 1985

HM84 Halpern, J.Y., Moses, Y.: Knowledge and Common Knowledge in a Distributed Environment, Proc. of the third ACM Conference on Principles of Distributed Computing, p. 50-61, 1984

HNS89 Hultman, J., Nyberg, A., Svensson, M.: A Software Architecture for Autonomous Systems, Research Report LiTH-IDA-R-89-40, Univ. Linköping, 1989

Huh87 Huhns, M.N.: Distributed Artificial Intelligence, Vol.I, Pitman/ Morgan Kaufmann Publ., San Mateo,CA, 1987

Huß91 Hußmann, M.: AGENTS Manual, 1991

KH81 Kornfeld, W.A., Hewitt, C.E.: The Scientific Community Metaphor, IEEE Transactions on Systems, Man and Cybernetics, SMC-11(1):24-33,Jan. 1981, also in (BG88)

KHH88 Kephart, J.O.,Hogg, T., Huberman, B.A.: Dynamics of Computational Ecosystems: Implications for DAI, in (GH88), 1988

KK90 Kowalski, R., Kim, J.-S.: A Metalogic Programming Approach to Multi-Agent Knowledge and Belief, Dept. of Computing, Imperail College of Science, Technology and Medicine, 180 Queen's Gate, London, 1990

LC81 Lesser, V.R., Corkill, D.D.: Functionally Accurate, Cooperative Distributed Systems, IEEE Transactions on Systems, Man and Cybernetics, SMC-11(1):81-96, January 1981, also in (BG88)

LC83 Lesser, V.R., Corkill, D.D.: The Distributed Vehicle Monitoring Testbed: A Tool for investigating Distributed Problem Solving Networks, AI Magazine 4(3): 15-33, Fall 1983

LC87 Lesser, V.R., Corkill, D.D.: Distributed Problem Solving, in Encyclopedia of Artificial Intelligence, Sharpio, S.C., p245-251, John Wiley and Sons, NY, 1987

Len75 Lenat, D.B.: BEINGs: Knowledge as Interacting Experts. Proceedings of the 1975 IJCAI, p 126-133, 1975

LE80 Lesser, V.R., Erman, L.D.:Distributed Interpretation: A Model and Experiment, IEEE Transactions on Computers C-29(12):1144-1163, Dec. 1980, also in (BG88)

Mae89 Maes, P.: The Dynamics of Action Selection, IJCAI89, 991-997, 1989

Mal87 Malone, T.W.: Modeling Coordination in Organizations and Markets, Management Science, 33(10):1317-1332, 1987

McCR87 James L.McClelland, David E.Rummelhart, and the PDP Research Group. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. MIT Press, Cambridge MA, 1987

MM88 Moyson, F., Manderick, B.: The collective Behavior of Ants: An Example of Self-Organization in Massive Parallelism, AI Memo 88-7, Vrije Universiteit Brussel, 1988

MSHI78 McCarthy, J., Sato, M., Hayashi, T., Igrashi, S.: On the Model Theory of Knowledge, Technical Report AIM-312, Stanford University 1978

Mül91 Müller, J.: From Silly Stand-Alone Systems to Cooperative Intelligent Agent Societies, in Proc. of the European Simulation Multiconference ESM91, Erik Mosekilde, editor, 1991

RGG86 Rosenschein, J.S., Ginsburg, M., Genesereth, M.R.: Cooperation without Communication., Proc. of AAAI86 , pages 51-57, 1986

Ric89 Richter, M.M.: Prinzipien der Künstlichen Intelligenz, Teubner Verlag, Stuttgart, 1989

SCHTW81 Steeb, R, Cammararta, S., Hayes-Roth, F.A., Thorndyke, P.W., Wesson, R.B.: Architectures for Distributed Intelligence for Air Fleet Control. Technical Report R-2728-ARPA, Rand Cooperation, Santa Monica CA, 1981, also in (BG88)

See91 Seel, N.: The 'Logical Omniscience' of Reactive Systems, in Proceedings of the 8th Conference of the Society for the Study of Artificial Intelligence and Simulation of Behavior, AISB91, April 1991

Sim81 Simon, A.H.: The Sciences of the Artificial, MIT Press, Cambridge, MA., 1981

Smi80 Smith, R.G.: The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver, IEEE Transactions on Computers, C-29(12):1104-1113, December 1980, also in BG88

Suc87 Suchmann, L.: Plans and Situated Actions: The Problem of Human-Machine Communication, Cambridge University Press, New York, 1987

Wer88a: Werner, E.: Toward a Theory of Communication and Cooperation for Multiagent Planning, RAK88, 129-143, 1988

Wer88b Werner, E.: Social Intensions, ECAI88, 719-723, 1988

Wit90 Wittig, T.: ARCHON - Cooperation of Heterogeneous On-Line Systems, ESPRIT-Project 2256, 1990

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
67608 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder per anonymem ftp von ftp.dfki.uni-kl.de (131.246.241.100) unter pub/Publications bezogen werden.
Die Berichte werden, wenn nicht anders gekenn-zeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all published papers so far are obtainable from the above address or per anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) under pub/Publications.
The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-92-45
Elisabeth André, Thomas Rist: The Design of Illustrated Documents as a Planning Task
21 pages

RR-92-46
Elisabeth André, Wolfgang Finkler, Winfried Graf, Thomas Rist, Anne Schauder, Wolfgang Wahlster: WIP: The Automatic Synthesis of Multimodal Presentations
19 pages

RR-92-47
Frank Bomarius: A Multi-Agent Approach towards Modeling Urban Traffic Scenarios
24 pages

RR-92-48
Bernhard Nebel, Jana Koehler:
Plan Modifications versus Plan Generation: A Complexity-Theoretic Perspective
15 pages

RR-92-49
Christoph Klauck, Ralf Legleitner, Ansgar Bernardi:
Heuristic Classification for Automated CAPP
15 pages

RR-92-50
Stephan Busemann:
Generierung natürlicher Sprache
61 Seiten

RR-92-51
Hans-Jürgen Bürckert, Werner Nutt:
On Abduction and Answer Generation through Constrained Resolution
20 pages

RR-92-52
Mathias Bauer, Susanne Biundo, Dietmar Dengler, Jana Koehler, Gabriele Paul: PHI - A Logic-Based Tool for Intelligent Help Systems
14 pages

RR-92-53
Werner Stephan, Susanne Biundo:
A New Logical Framework for Deductive Planning
15 pages

RR-92-54
Harold Boley: A Direkt Semantic Characterization of RELFUN
30 pages

RR-92-55
John Nerbonne, Joachim Laubsch, Abdel Kader Diagne, Stephan Oepen: Natural Language Semantics and Compiler Technology
17 pages

RR-92-56
Armin Laux: Integrating a Modal Logic of Knowledge into Terminological Logics
34 pages

RR-92-58
Franz Baader, Bernhard Hollunder:
How to Prefer More Specific Defaults in Terminological Default Logic
31 pages

RR-92-59
Karl Schlechta and David Makinson: On Principles and Problems of Defeasible Inheritance
13 pages

RR-92-60
Karl Schlechta: Defaults, Preorder Semantics and Circumscription
19 pages

RR-93-02
Wolfgang Wahlster, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, Thomas Rist:
Plan-based Integration of Natural Language and Graphics Generation
50 pages

RR-93-03
Franz Baader, Berhard Hollunder, Bernhard Nebel, Hans-Jürgen Profitlich, Enrico Franconi: An Empirical Analysis of Optimization Techniques for Terminological Representation Systems
28 pages

RR-93-04
Christoph Klauck, Johannes Schwagereit: GGD: Graph Grammar Developer for features in CAD/CAM
13 pages

RR-93-05
Franz Baader, Klaus Schulz: Combination Tech-niques and Decision Problems for Disunification
29 pages

RR-93-06
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: On Skolemization in Constrained Logics
40 pages

RR-93-07
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: Concept Logics with Function Symbols
36 pages

RR-93-08
Harold Boley, Philipp Hanschke, Knut Hinkelmann, Manfred Meyer: COLAB: A Hybrid Knowledge Representation and Compilation Laboratory
64 pages

RR-93-09
Philipp Hanschke, Jörg Würtz: Satisfiability of the Smallest Binary Program
8 Seiten

RR-93-10
Martin Buchheit, Francesco M. Donini, Andrea Schaerf: Decidable Reasoning in Terminological Knowledge Representation Systems
35 pages

RR-93-11
Bernhard Nebel, Hans-Juergen Buerckert: Reasoning about Temporal Relations: A Maximal Tractable Subclass of Allen's Interval Algebra
28 pages

RR-93-12
Pierre Sablayrolles: A Two-Level Semantics for French Expressions of Motion
51 pages

RR-93-13
Franz Baader, Karl Schlechta: A Semantics for Open Normal Defaults via a Modified Preferential Approach
25 pages

RR-93-14
Joachim Niehren, Andreas Podelski,Ralf Treinen: Equational and Membership Constraints for Infinite Trees
33 pages

RR-93-15
Frank Berger, Thomas Fehrle, Kristof Klöckner, Volker Schölles, Markus A. Thies, Wolfgang Wahlster: PLUS - Plan-based User Support
Final Project Report
33 pages

RR-93-16
Gert Smolka, Martin Henz, Jörg Würtz: Object-Oriented Concurrent Constraint Programming in Oz
17 pages

RR-93-17
Rolf Backofen: Regular Path Expressions in Feature Logic
37 pages

RR-93-18
Klaus Schild: Terminological Cycles and the Propositional μ-Calculus
32 pages

RR-93-20
Franz Baader, Bernhard Hollunder: Embedding Defaults into Terminological Knowledge Representation Formalisms
34 pages

RR-93-22
Manfred Meyer, Jörg Müller: Weak Looking-Ahead and its Application in Computer-Aided Process Planning
17 pages

RR-93-23
Andreas Dengel, Ottmar Lutzy: Comparative Study of Connectionist Simulators
20 pages

RR-93-24
Rainer Hoch, Andreas Dengel: Document Highlighting — Message Classification in Printed Business Letters
17 pages

RR-93-25
Klaus Fischer, Norbert Kuhn: A DAI Approach to Modeling the Transportation Domain
93 pages

RR-93-26
Jörg P. Müller, Markus Pischel: The Agent Architecture InteRRaP: Concept and Application
99 pages

RR-93-27
Hans-Ulrich Krieger: Derivation Without Lexical Rules
33 pages

RR-93-28
Hans-Ulrich Krieger, John Nerbonne, Hannes Pirker: Feature-Based Allomorphy
8 pages

RR-93-29
Armin Laux: Representing Belief in Multi-Agent Worlds viaTerminological Logics
35 pages

RR-93-33
Bernhard Nebel, Jana Koehler:
Plan Reuse versus Plan Generation: A Theoretical and Empirical Analysis
33 pages

RR-93-34
Wolfgang Wahlster:
Verbmobil Translation of Face-To-Face Dialogs
10 pages

RR-93-35
Harold Boley, François Bry, Ulrich Geske (Eds.): Neuere Entwicklungen der deklarativen KI-Programmierung — *Proceedings*
150 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

RR-93-36
Michael M. Richter, Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Gabriele Schmidt: Von IDA bis IMCOD: Expertensysteme im CIM-Umfeld
13 Seiten

RR-93-38
Stephan Baumann: Document Recognition of Printed Scores and Transformation into MIDI
24 pages

RR-93-40
Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt, Andrea Schaerf: Queries, Rules and Definitions as Epistemic Statements in Concept Languages
23 pages

RR-93-41
Winfried H. Graf: LAYLAB: A Constraint-Based Layout Manager for Multimedia Presentations
9 pages

RR-93-42
Hubert Comon, Ralf Treinen:
The First-Order Theory of Lexicographic Path Orderings is Undecidable
9 pages

RR-93-44
Martin Buchheit, Manfred A. Jeusfeld, Werner Nutt, Martin Staudt: Subsumption between Queries to Object-Oriented Databases
36 pages

RR-93-45
Rainer Hoch: On Virtual Partitioning of Large Dictionaries for Contextual Post-Processing to Improve Character Recognition
21 pages

DFKI Technical Memos

TM-92-01
Lijuan Zhang: Entwurf und Implementierung eines Compilers zur Transformation von Werkstückrepräsentationen
34 Seiten

TM-92-02
Achim Schupeta: Organizing Communication and Introspection in a Multi-Agent Blocksworld
32 pages

TM-92-03
Mona Singh:
A Cognitiv Analysis of Event Structure
21 pages

TM-92-04
Jürgen Müller, Jörg Müller, Markus Pischel, Ralf Scheidhauer:
On the Representation of Temporal Knowledge
61 pages

TM-92-05
Franz Schmalhofer, Christoph Globig, Jörg Thoben:
The refitting of plans by a human expert
10 pages

TM-92-06
Otto Kühn, Franz Schmalhofer: Hierarchical skeletal plan refinement: Task- and inference structures
14 pages

TM-92-08
Anne Kilger: Realization of Tree Adjoining Grammars with Unification
27 pages

TM-93-01
Otto Kühn, Andreas Birk: Reconstructive Integrated Explanation of Lathe Production Plans
20 pages

TM-93-02
Pierre Sablayrolles, Achim Schupeta:
Conlfict Resolving Negotiation for COoperative Schedule Management
21 pages

TM-93-03
Harold Boley, Ulrich Buhrmann, Christof Kremer:
Konzeption einer deklarativen Wissensbasis über recyclingrelevante Materialien
11 pages

TM-93-04
Hans-Günther Hein: Propagation Techniques in WAM-based Architectures — The FIDO-III Approach
105 pages

DFKI Documents

D-92-24
Jürgen Müller, Donald Steiner (Hrsg.):
Kooperierende Agenten
78 Seiten

D-92-25
Martin Buchheit: Klassische Kommunikations- und Koordinationsmodelle
31 Seiten

D-92-26
Enno Tolzmann:
Realisierung eines Werkzeugauswahlmoduls mit Hilfe des Constraint-Systems CONTAX
28 Seiten

D-92-27
Martin Harm, Knut Hinkelmann, Thomas Labisch: Integrating Top-down and Bottom-up Reasoning in COLAB
40 pages

D-92-28
Klaus-Peter Gores, Rainer Bleisinger: Ein Modell zur Repräsentation von Nachrichtentypen
56 Seiten

D-93-01
Philipp Hanschke, Thom Frühwirth:
Terminological Reasoning with Constraint Handling Rules
12 pages

D-93-02
Gabriele Schmidt, Frank Peters, Gernod Laufkötter: User Manual of COKAM+
23 pages

D-93-03
Stephan Busemann, Karin Harbusch(Eds.):
DFKI Workshop on Natural Language Systems: Reusability and Modularity - Proceedings
74 pages

D-93-04
DFKI Wissenschaftlich-Technischer Jahresbericht 1992
194 Seiten

D-93-05
Elisabeth André, Winfried Graf, Jochen Heinsohn, Bernhard Nebel, Hans-Jürgen Profitlich, Thomas Rist, Wolfgang Wahlster:
PPP: Personalized Plan-Based Presenter
70 pages

D-93-06
Jürgen Müller (Hrsg.):
Beiträge zum Gründungsworkshop der Fachgruppe Verteilte Künstliche Intelligenz Saarbrücken 29.-30. April 1993
235 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-07
Klaus-Peter Gores, Rainer Bleisinger:
Ein erwartungsgesteuerter Koordinator zur partiellen Textanalyse
53 Seiten

D-93-08
Thomas Kieninger, Rainer Hoch: Ein Generator mit Anfragesystem für strukturierte Wörterbücher zur Unterstützung von Texterkennung und Textanalyse
125 Seiten

D-93-09
Hans-Ulrich Krieger, Ulrich Schäfer:
TDL ExtraLight User's Guide
35 pages

D-93-10
Elizabeth Hinkelman, Markus Vonerden,Christoph Jung: Natural Language Software Registry
(Second Edition)
174 pages

D-93-11
Knut Hinkelmann, Armin Laux (Eds.):
DFKI Workshop on Knowledge Representation Techniques — Proceedings
88 pages

D-93-12
Harold Boley, Klaus Elsbernd, Michael Herfert, Michael Sintek, Werner Stein:
RELFUN Guide: Programming with Relations and Functions Made Easy
86 pages

D-93-14
Manfred Meyer (Ed.): Constraint Processing – Proceedings of the International Workshop at CSAM'93, July 20-21, 1993
264 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-15
Robert Laux: Untersuchung maschineller Lernverfahren und heuristischer Methoden im Hinblick auf deren Kombination zur Unterstützung eines Chart-Parsers
86 Seiten

D-93-20
Bernhard Herbig:
Eine homogene Implementierungsebene für einen hybriden Wissensrepräsentationsformalismus
97 Seiten

D-93-21
Dennis Drollinger:
Intelligentes Backtracking in Inferenzsystemen am Beispiel Terminologischer Logiken
53 Seiten